

DIGITAL COMMUNICATION SYSTEMS HAVING DECREASED MEMORY

BACKGROUND

This invention relates to telecommunication systems and more particularly to
5 methods and apparatus requiring less memory in radio systems, such as a wideband
code division multiple access (WCDMA) system.

Digital communication systems include time-division multiple access (TDMA)
systems, such as cellular radio telephone systems that comply with the GSM
telecommunication standard and its enhancements like GSM/EDGE, and code-division
10 multiple access (CDMA) systems, such as cellular radio telephone systems that comply
with the IS-95, cdma2000, and WCDMA telecommunication standards. Digital
communication systems also include "blended" TDMA and CDMA systems, such as
cellular radio telephone systems that comply with the universal mobile
telecommunications system (UMTS) standard, which specifies a third generation (3G)
15 mobile system being developed by the European Telecommunications Standards
Institute (ETSI) within the International Telecommunication Union's (ITU's) IMT-2000
framework. The Third Generation Partnership Project (3GPP) promulgates the UMTS
standard. This application focusses on WCDMA systems for simplicity, but it will be
understood that the principles described in this application can be implemented in other
20 digital communication systems.

FIG. 1 depicts a communication system such as a WCDMA system that includes
a base station (BS) 10 handling connections with four mobile stations (MSs) 1, 2, 3, 4
that each use downlink (i.e., base-to-mobile or forward) and uplink (i.e., mobile-to-base
or reverse) channels. In the downlink, BS 10 transmits to each mobile at a respective
25 power level, and the signals transmitted by BS 10 are spread using orthogonal code
words. In the uplink, MS 1 - MS 4 transmit to BS 10 at respective power levels.
Although not shown, BS 10 also communicates with a radio network controller (RNC),
which in turn communicates with a public switched telephone network (PSTN).

WCDMA is based on direct-sequence spread-spectrum techniques. Two
30 different codes are used for separating base stations and physical channels in the
downlink (base-to-terminal) direction. Scrambling codes are pseudo-noise (pn)
sequences that are mainly used for separating the base stations or cells from each
other. Channelization codes are orthogonal sequences that are used for separating
different physical channels (terminals or users) in each cell or under each scrambling

code. User terminals communicate with the system through, for example, respective dedicated physical channels (DPCHs). It will be appreciated that WCDMA terminology is used here and that other systems have corresponding terminology. Scrambling and channelization codes are well known in the art.

5 The signals transmitted in the exemplary WCDMA system depicted in FIG. 1 can be formed with a transmitter system 100 such as that depicted in FIG. 2. An information data stream to be transmitted, such as a serial stream of bits to be sent on a downlink physical channel, is converted from serial form to parallel form by a suitable converter 102, and the now-parallel bits to be sent are provided to a modulation mapper 104 that transforms groups of bits into respective modulation symbols. WCDMA and other transmitters employ quadrature modulation, so the mapper 104 generates streams of in-phase (I) and quadrature (Q) symbols that are provided to respective multipliers 106, 108 that combine the symbols with a selected channelization code. As shown in FIG. 2, the resultant streams are provided to a combiner 110, and then the resultant combined stream is provided to another multiplier 112 that combines the combined stream with a selected scrambling code. The resultant channelized, scrambled signal may then be further processed and impressed on a suitable carrier signal (not shown) for transmission.

20 The multiplications are usually carried out by exclusive-OR operations, and the information data stream and the scrambling code can have the same or different bit rates. Each information data stream or channel is allocated a unique channelization code, and a plurality of coded information signals simultaneously modulate a radio-frequency carrier signal.

25 In order to accommodate the increasing demand for higher data rates in wireless user equipment (UE), such as cellular telephones, combination cellular telephones-personal digital assistants, and wireless-enabled personal computers, a high-speed downlink shared channel (HS-DSCH) was introduced in WCDMA. The HS-DSCH has a spreading factor of sixteen and can use several channelization codes simultaneously, with modulation being either quadrature phase shift keying (QPSK) or 16-ary quadrature amplitude modulation (16QAM). Each transmission time interval (TTI) includes one transport block, and the length of a TTI is three slots. After encoding, interleaving, and rate matching, the bits to be transmitted are distributed over one or more channelization codes. This is described in "Multiplexing and channel coding (FDD)", 3GPP Technical Specification (TS) 25.212 ver. 5.6.0 (Sept. 2003), for example.

Since the chip rate in a direct-sequence CDMA system is typically constant, a higher spreading factor generally corresponds to a lower information bit-rate.

The transmitter 100 depicted in FIG. 2 is suitable for the HS-DSCH, and the 16QAM symbol constellation generated by the modulation mapper 104 is depicted in FIG. 3. The region marked i_1 is where the first bit is mapped when it is one; the region marked q_1 is where the second bit is mapped when it is one; the region marked i_2 is where the third bit is mapped when it is one; and the region marked q_2 is where the fourth bit is mapped when it is one. It should be noted that input binary zeroes are mapped to ones and input binary ones are mapped to negative ones.

At a mobile station or other receiver, the modulated carrier signal is processed to produce an estimate of the original information data stream intended for the receiver. This process is known as demodulation. The composite received baseband spread signal is commonly provided to a rake processor that includes a number of "fingers", or de-spreaders, that are each assigned to respective ones of selected components, such as multipath echoes or images, in the received signal. Each finger combines a received component with the scrambling sequence and the channelization code so as to de-spread the received composite signal. The rake processor typically de-spreads both sent information data and pilot or training symbols that are included in the composite signal.

Various aspects of rake receivers are described in G. Turin, "Introduction to Spread-Spectrum Antimultipath Techniques and Their Application to Urban Digital Radio", Proc. IEEE, vol. 68, pp. 328-353 (March 1980); U.S. Patent No. 5,305,349 to Dent for "Quantized Coherent Rake Receiver"; U.S. Patent Application Publication No. 2001/0028677 by Wang et al. for "Apparatus and Methods for Finger Delay Selection in Rake Receivers"; and U.S. Patent Applications No. 09/165,647 filed on October 2, 1998, by G. Bottomley for "Method and Apparatus for Interference Cancellation in a Rake Receiver" and No. 09/344,898 filed on June 25, 1999, by Wang et al. for "Multi-Stage Rake Combining Methods and Apparatus".

FIG. 4 is a block diagram of a typical HS-DSCH receiver 400. A rake processor 402 receives over-sampled chip values from one or more analog-to-digital converters (not shown). The rake processor 402 sub-samples the received over-sampled chip sequence and multiplies each chip with the corresponding scrambling and channelization codes. The first chip to select is determined by the appropriate multi-path delay, and these delays are determined by a suitable path estimator 404. The

output of the rake processor is a set of symbols corresponding to different channelization codes and multi-path delays. It will be understood that FIG. 4 omits any components that might be used for recovering the baseband modulation of a radio-frequency carrier signal.

5 A combiner 406 multiplies the symbols for each delay with the complex conjugate of an estimate of the impulse response of the propagation channel for the corresponding delay and sums the contributions over the delays. The channel estimates are produced by a suitable channel estimator 408. The combiner 406 thus reproduces the modulated complex symbols that were transmitted. Not shown in
10 FIG. 4 are various registers or memories that are usually used in practice to enable received signals to be processed while new signals are received. For example, a memory is placed before the rake processor 402 for so-called over-sampled chip buffering or after the rake processor for so-called symbol buffering.

 When receiving 16QAM-modulated symbols, the constellation of which is
15 depicted in FIG. 3, the receiver has to determine the distance d between the Q-axis and the point on the I-axis that is half-way between the first and second columns of constellation points in the right half-plane in order to decide which symbols have been received. As depicted in FIG. 3, the first column of points is located at 0.4472 on the I-axis and the second column of points is located at 1.3416, so the distance d is 0.8944.
20 Because of the symmetry of the constellation, the determined distance d separates all of the constellation points into a uniform pattern of squares.

 The distance d is proportional to the amplitude of the received signal, and that amplitude varies with time due to signal fading in the propagation channel. Thus, when the receiver is in a fading dip, the distance d is small, and when the receiver is on a
25 fading peak, the distance d is large. Accordingly, the receiver must regularly update its determination of the distance d . FIG. 4 shows a memory 410 after the combiner 406 that stores a number of complex symbols generated by the combiner. A decision boundary estimator 412 computes the distance d separately for each channelization code based on respective symbols in the memory 410 and on respective previously
30 computed values of the distance d for each channelization code. Using the estimates of the distance d computed by the estimator 412, a soft bit value estimator 414 computes so-called soft bit values from the complex symbols stored in the memory 410.

 The computed soft bit values may be stored in a memory called a Hybrid ARQ (HARQ) buffer that implements hybrid automatic repeat request (HARQ). HARQ is a

combination (a hybrid) of ARQ and forward error correction coding (FEC), which are well known techniques for increasing the probability of successful signal reception in the face of channel noise and interference from other users. Another known error handling technique is interleaving, in which symbols are interleaved, or shuffled out of order, at the transmitter and correspondingly deinterleaved, or re-ordered, at the receiver. HARQ, interleaving, and the HS-DSCH are described in, for example, the 3GPP TS 25.212 ver. 5.6.0 (Sept. 2003) standard cited above.

FIG. 4 depicts the HARQ and deinterleaving functionality in a combined deinterleaver/HARQ buffer 416. The computed soft bit values are not removed from the HARQ buffer 416 until the transmitted transport block has been decoded correctly by a suitable decoder 418 or when higher signalling layers inform the HARQ buffer to clear the soft bit values. Multiple retransmissions may occur for a given block until it has been decoded correctly. The soft bit values from each retransmission are combined and stored in the HARQ buffer 416. The transmitter, such as a cellular base station, is eventually informed, for example via an uplink high-speed dedicated physical control channel (HS-DPCCH), that the block has been received correctly.

Having the memory 410 between the combiner 406 and the decision boundary estimator 412 as in FIG. 4 is costly in several ways. For example, the memory size must be on the order of five kilobits (5 kbits), and this requires a corresponding area on one of the integrated circuit (IC) chips used for the receiver 400. The chip die area could be reduced if the memory 410 could be eliminated.

SUMMARY

Applicants have developed methods and apparatus that do not require a memory between the combiner and decision boundary estimator and that enjoy unaltered performance. Applicants' methods and apparatus include new ways of estimating the decision boundary and may also include new ways of estimating soft bit values.

In one aspect of Applicants' invention, a receiver includes a path estimator that selects at least one multi-path delay for received complex symbols; a rake processor that multiplies chips in a chip sequence with at least a corresponding channelization code based on at least one multi-path delay selected by the path estimator and generates a set of symbols corresponding to the channelization codes and multi-path delays; a channel estimator that generates estimates of an impulse response of the

propagation channel for the corresponding multi-path delay; and a combiner that multiplies symbols for a multi-path delay with complex conjugates of corresponding impulse response estimates generated by the channel estimator and reproduces the modulated symbols that were transmitted. The receiver further includes a decision
5 boundary estimator/latch controller that generates estimates of a decision boundary based on reproduced modulated symbols; a latch that is responsive to the decision boundary estimator/latch controller; a soft bit value estimator that generates, for respective channelization codes, estimates of bit values of complex symbols based on the decision boundary estimates; and a decoder. When the latch is in a first state, the
10 decision boundary estimator generates decision boundary estimates and the soft bit value estimator does not generate soft bit values, and when the latch is in a second state, the decision boundary estimator generates decision boundary estimates and the soft bit value estimator generates soft bit values. Decision boundary estimates used for symbols from a channelization code are generated from symbols from at least one
15 different channelization code.

In another aspect of the invention, a method in a receiver of determining decision boundary estimates based on received symbols from one or more channelization codes includes the step of determining a decision boundary for a respective channelization
code using received symbols for at least one other channelization code and at least one
20 decision boundary estimate determined for another channelization code.

In another aspect of the invention, a method of decision boundary estimation and soft bit value estimation includes the steps of running a combiner for M symbols of a channelization code n; generating soft bit value estimates from an output of the combiner; determining a sample decision boundary estimate based on the M symbols;

25 and updating the sample decision boundary estimate based on a variable $d_{\Sigma}^{(n)}$ that sums a sample decision boundary $\hat{d}^{(n)}$ over a time interval corresponding to M · N symbols and on a variable $d_{\text{filter}, l}$ that reflects a history of the variable $d_{\Sigma}^{(n)}$.

In still another aspect of the invention, a computer-readable medium contains a computer program for determining decision boundary estimates based on received
30 symbols from one or more channelization codes in a receiver. The computer program performs the step of determining a decision boundary for a respective channelization code using received symbols for at least one other channelization code and at least one decision boundary estimate determined for another channelization code.

BRIEF DESCRIPTION OF THE DRAWINGS

The objects and advantages of Applicants' invention will be apparent from reading this description in conjunction with the drawings, in which:

5 FIG. 1 depicts a typical communication system;

FIG. 2 is a block diagram of a typical transmitter suitable for the communication system;

FIG. 3 depicts a 16QAM signal constellation and mapping;

10 FIG. 4 is a block diagram of a typical receiver for a high-speed downlink shared channel;

FIG. 5 is a block diagram of an exemplary embodiment of Applicants' receiver for a high-speed downlink shared channel;

FIG. 6 is a flow chart of a method of decision boundary estimation, latch control, and soft bit value estimation; and

15 FIG. 7 is a flow chart of another method of decision boundary estimation, latch control, and soft bit value estimation.

DETAILED DESCRIPTION

FIG. 5 is a block diagram of an exemplary embodiment of Applicants' receiver 20 500 for a channel such as the HS-DSCH that has advantageously low memory. Such a receiver can be implemented in the base band IC chip of a UMTS terminal or even in software executed by a suitable processor. Applicants' exemplary receiver 500 has a rake processor 502, a path estimator 504, a combiner 506, a channel estimator 508, a soft bit value estimator 514, a deinterleaver/HARQ buffer 516, and a decoder 518 that 25 are comparable to the like components of the receiver 400 depicted in FIG. 4.

Applicants' receiver 500 differs from the receiver 400 in that it has no memory comparable to the memory 410. Applicants' receiver 500 includes a decision boundary estimator/latch controller 511 and a latch 513, which determines when the receiver 500 computes the decision boundary estimate without generating soft values, and when the 30 receiver 500 computes the decision boundary estimate and at the same time generates soft bit values. The operation of the estimator/controller 511 and latch 513 may be understood from an example, in which it is assumed that the receiver 500 is receiving a plurality N of channelization codes, which may be respectively numbered $n = 0, 1, \dots, N-1$, and that the receiver updates its estimate of the decision boundary every $M \cdot N$

symbols or equivalently every M symbols for each channelization code. It will be understood that the numbering of the N channelization codes is arbitrary.

To start the example, the latch 513 is set such that the decision boundary estimator estimates the boundary for a channelization code, say, code $n = 0$, and such that the soft bit value estimator 514 does not generate soft bit values. This is to say that the latch 513 is open as shown in FIG. 5. Thus, the receiver 500 runs only either the rake processor 502 and combiner 506 or just the combiner 506, depending on whether the receiver implements either over-sampled chip buffering or symbol buffering, and their associated components, and the decision boundary estimator 511, which computes an estimate.

Continuing the example, the rake processor 502 and combiner 506 or just the combiner 506 are/is rerun for the channelization code $n = 0$ using the previously computed decision boundary estimate. Next, the combiner 506 is run for another channelization code, say, code $n = 1$, and soft bit value estimates are generated by the estimator 514 based on the decision boundary estimate computed for the channelization code $n = 0$. This is to say that the latch 513 is closed. At the same time, a new decision boundary estimate using the channelization code $n = 1$ symbols is computed by the estimator/controller 511. This procedure is repeated for each remaining channelization code n' , i.e., soft bit value estimates are generated based on the decision boundary estimate computed for channelization code $n'-1$ and a decision boundary estimate is computed for channelization code n' , until all channelization codes have been processed.

It can thus be understood that Applicants' receiver 500 implements a method in which one or more decision boundary estimates based on symbols from one or more channelization codes are used for soft bit value estimation for a different channelization code. Simulation results show that Applicants' receiver has substantially the same performance as a receiver that implements a method in which the decision boundary estimate based on symbols from one channelization code is used for soft bit value estimation for that same channelization code.

FIG. 6 is a flow chart of a method that can be implemented by the receiver 500 in the decision boundary estimator/latch controller 511, latch 513, and soft bit value estimator 514. In step 601, M complex symbols s are read from the combiner 506 for a channelization code, say, code $n = 0$. Such reading usually involves running the combiner 506 or the rake processor 502 and the combiner 506, depending on the type

of buffering, if any, as noted above. It should be observed that reading a channelization code such as $n = 0$ twice in a row entails reading the same M complex symbols from the combiner 506. In the case in which there is only one channelization code in use, i.e., $N = 1$, the M symbols read from the combiner 506 are incremented after the

5 second time they are read.

Step 602 determines whether the first channelization code, say, code $n = 0$, is being processed. If the first channelization code is being processed, then whether the latch 513 is open is determined (step 603). If the latch is not open, there is no decision boundary estimate available to generate soft bit value estimates because this is the first
10 code and an estimate has not yet been computed. Thus, the latch 513 is opened by the estimator/controller 511 (step 604), and the combiner 506 is run for M symbols (step 601) in order to enable computation of a sample decision boundary estimate by the estimator/controller (see step 607), without computation of soft bit value estimates by the estimator 514. If the latch 513 is determined in step 603 to be open, the latch is
15 closed and soft bit value estimates are generated (step 605) by the estimator 514 using the decision boundary estimate available from the estimator/controller 511.

If step 602 determines that the first channelization code is not being processed, soft bit value estimates are generated based on the last available decision boundary estimate in step 606. Soft bit value estimates can, for example, be generated in steps
20 605, 606 as follows. Let s be a complex symbol representing four soft bit values b_1, b_2, b_3, b_4 and let d be the decision boundary. Then, the soft bit value estimates are given by:

$$\hat{b}_1 = \text{Re } s$$

$$\hat{b}_2 = \text{Im } s$$

$$\hat{b}_3 = \rho(d - \text{Re } s) \text{ for } \text{Re } s > 0$$

$$\hat{b}_3 = \rho(d + \text{Re } s) \text{ for } \text{Re } s \leq 0$$

$$\hat{b}_4 = \rho(d - \text{Im } s) \text{ for } \text{Im } s > 0$$

$$\hat{b}_4 = \rho(d + \text{Im } s) \text{ for } \text{Im } s \leq 0$$

where $\text{Re } s$ is the real part of s , $\text{Im } s$ is the imaginary part of s , and ρ is a factor that
30 compensates for the limited resolutions, or bit widths, of the soft bit value estimates $\hat{b}_1, \hat{b}_2, \hat{b}_3, \hat{b}_4$ in the computations and for the better reliability of \hat{b}_1, \hat{b}_2 , which do not depend on an estimate of the decision boundary d . It is currently believed that suitable

values of ρ are $0 < \rho \leq 1$ and that a particularly suitable value is $\rho = 0.75$. After steps 605, 606, the process flow returns to step 601, and continues for the next channelization code $n+1$ until all N codes have been processed.

5 In step 607, a sample decision boundary $\hat{d}^{(n)}$ based on M complex symbols s from a channelization code n can be computed according to, for example, the following:

$$\hat{d}^{(n)} = \frac{\kappa}{2M} \sum_{m=1}^M |\text{Re } s_m| + |\text{Im } s_m|$$

10 where κ is a factor that compensates for folding in noise that causes over-estimating the sample decision boundary $\hat{d}^{(n)}$. It is currently believed that suitable values of κ are $0 < \kappa \leq 1$ and that a particularly suitable value is $\kappa = 0.95$. It will be noted that the decision boundary calculations (step 607 and 608) by the estimator/controller 511 can be performed in parallel or serial with the latch controlling the soft bit value estimations (step 602-606) by the controller 511 and the estimator 514.

15 In step 608, the sample decision boundary is updated, for example in the following way. Let $d_{\Sigma}^{(n)}$ be a temporary variable that sums sample decision boundaries $\hat{d}^{(n)}$ over a time interval corresponding to $M \cdot N$ complex symbols. The variable $d_{\Sigma}^{(n)}$ is given by:

$$d_{\Sigma}^{(n)} = \begin{cases} \hat{d}^{(0)}, n = 0 \\ d_{\Sigma}^{(n-1)} + \hat{d}^{(n)}, n \neq 0 \end{cases}$$

20 where n is the index identifying the N channelization codes as used above. Also let $d_{\text{filter}, l}$ reflect the history of the variable $d_{\Sigma}^{(n)}$ in a manner that will be clear from the following. The decision boundary estimate d_0 to be applied for the code $n = 0$ in the soft bit value estimation is:

$$d_0 = (1 - \alpha)d_{\text{filter}, l} + \alpha d_{\Sigma}^{(0)}$$

where α is a filter parameter. The decision boundary estimates d_n to be applied for the codes $n = 1, 2, \dots, N-1$ in the soft bit value estimation are:

25
$$d_n = (1 - \alpha)d_{\text{filter}, l} + \alpha \frac{1}{n} d_{\Sigma}^{(n-1)}.$$

It will be seen that the sample decision boundary $\hat{d}^{(n)}$ is included in the decision boundary estimate d_n only for the first channelization code $n = 0$ and that the other codes n do not include the sample decision boundary $\hat{d}^{(n)}$ for channelization code n in

their decision boundaries d_n . Once sample decision boundaries $\hat{d}^{(n)}$ have been computed for all of the N codes, $d_{\text{filter},l}$ is updated according to:

$$d_{\text{filter},l+1} = (1 - \alpha)d_{\text{filter},l} + \alpha \frac{1}{N} d_{\Sigma}^{(N)}.$$

The filter parameter α is advantageously selected as approximately

5 $\min(c f_D \cdot M / 160 \cdot 1500, 1)$ when this update is not the first update after a transmission gap and where f_D is the Doppler frequency shift at the receiver. It is currently believed that suitable values of the factor c are $1 \leq c \leq 10$ and that $c = 4$ is particularly suitable. The length of a transmission gap is given by $\text{TTI}_{\text{inter}}$, which tells the UE that every $\text{TTI}_{\text{inter}}$ transmission time interval is destined for the UE. If this update is the first update after a
10 transmission gap, the filter parameter α is advantageously selected as approximately $\min(c f_D \cdot M / 160 \cdot 1500 + 3 \text{TTI}_{\text{inter}} / 1500, 1)$. As noted above, it is currently believed that suitable values of the factor c are $1 \leq c \leq 10$ and that $c = 4$ is particularly suitable.

Nevertheless, if the transmitter, such as a cellular base station, changes its transmit power on the HS-DSCH in the next TTI, a suitable filter parameter is $\alpha = 1$,
15 which is to say that no old values of the decision boundary from the last TTI can be saved. In a WCDMA system the transmitter informs the UE of power changes on the HS-DSCH from one TTI to the next. The Doppler frequency shift f_D can be estimated in the UE, such as a mobile receiver, by any of many known methods. It is currently believed that suitable values of M are $40 \leq M \leq 320$ and that choosing $M = 160$ gives
20 good performance for transmitter-receiver relative speeds up to 150 km/h.

In an alternative embodiment, the method illustrated by FIG. 7 can be used for implementation of the decision boundary estimation, latch control, and soft bit value estimation functionality. In this method, there is no need to rerun the combiner 506 for the channelization code $n = 0$.

25 In step 701, the combiner 506 is run for M symbols of channelization code n , and in step 702, soft bit value estimates are generated by the estimator 514 from the output of the combiner. In step 703, a decision boundary estimate based on the M symbols is determined. In step 704, the decision boundary estimate is updated as follows, with the process flow returning to step 701 after updating, and continuing for the next
30 channelization code $n+1$ until all N codes have been processed.

Let $d_{\Sigma}^{(n)}$ be a temporary variable that sums the sample decision boundary $\hat{d}^{(n)}$ over a time interval corresponding to $M \cdot N$ complex symbols, with $d_{\Sigma}^{(n)}$ given by:

$$d_{\Sigma}^{(n)} = \begin{cases} 0, n = 0 \\ d_{\Sigma}^{(n-1)} + \hat{d}^{(n-1)}, n \neq 0 \end{cases}$$

Also let $d_{\text{filter}, l}$ reflect the history of the variable $d_{\Sigma}^{(n)}$. The decision boundary estimate d_0 to be applied for channelization code $n = 0$ is $d_0 = d_{\text{filter}, l}$, and the decision boundary estimates to be applied for the codes $n = 1, 2, \dots, N-1$ are:

5
$$d_n = (1 - \alpha)d_{\text{filter}, l} + \alpha \frac{1}{n} d_{\Sigma}^{(n-1)}.$$

Once sample decision boundaries $\hat{d}^{(n)}$ have been computed for all N channelization codes, the variable $d_{\text{filter}, l}$ is updated according to:

$$d_{\text{filter}, l+1} = (1 - \alpha)d_{\text{filter}, l} + \alpha \frac{1}{N} d_{\Sigma}^{(N)}.$$

As in the method described above in connection with FIG. 6, the filter parameter α is
10 selected as approximately $\min(cf_D \cdot M/160 \cdot 1500, 1)$ when this is not the first update after a transmission gap and as approximately $\min(cf_D \cdot M/160 \cdot 1500 + 3TTI_{\text{inter}}/1500, 1)$ when this is the first update after a transmission gap. Also as described above, it is currently believed that suitable values of the factor c are $1 \leq c \leq 10$, that $c = 4$ is particularly suitable, that $\alpha = 1$ if the transmitter changes the power on the HS-DSCH in the next
15 TTI, and that $M = 160$ gives good performance for transmitter-receiver relative speeds up to about 150 km/h.

In practice, the algorithm depicted in FIG. 7 may entail running the combiner 506, for the very first symbols that are received for the first channelization code $n = 0$, without computing soft bit values, so as to initialize $d_{\text{filter}, l}$ in step 704. It is also currently
20 believed that the algorithm in FIG. 7 is particularly advantageous when TTI_{inter} is unity and the Doppler frequency shift is low, e.g., $f_D < 30$ Hz.

In yet another alternative embodiment, steps 607 and 608 may be carried out only for channelization codes $n = 0, 1, \dots, N_{\text{red}}-1$, where the number N_{red} of channelization codes used for a decision boundary is less than or equal to the total
25 number N of channelization codes.

It will be appreciated that the procedures described above are carried out repetitively as necessary to respond to the time-varying characteristics of the channel between the transmitter and receiver. To facilitate understanding, many aspects of the invention are described in terms of sequences of actions to be performed by, for
30 example, elements of a programmable computer system. It will be recognized that the various actions could be performed by specialized circuits (e.g., discrete logic gates

interconnected to perform a specialized function or application-specific integrated circuits), by program instructions executed by one or more processors, or by a combination of both.

Moreover, the invention can additionally be considered to be embodied entirely
5 within any form of computer-readable storage medium having stored therein an appropriate set of instructions for use by or in connection with an instruction-execution system, apparatus, or device, such as a computer-based system, processor-containing system, or other system that can fetch instructions from a medium and execute the instructions. As used here, a "computer-readable medium" can be any means that can
10 contain, store, communicate, propagate, or transport the program for use by or in connection with the instruction-execution system, apparatus, or device. The computer-readable medium can be, for example but not limited to, an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system, apparatus, device, or propagation medium. More specific examples (a non-exhaustive list) of the computer-
15 readable medium include an electrical connection having one or more wires, a portable computer diskette, a random access memory (RAM), a read only memory (ROM), an erasable programmable read only memory (EPROM or Flash memory), an optical fiber, and a portable compact disc read only memory (CD-ROM).

Thus, the invention may be embodied in many different forms, not all of which
20 are described above, and all such forms are contemplated to be within the scope of the invention. For each of the various aspects of the invention, any such form may be referred to as "logic configured to" perform a described action, or alternatively as "logic that" performs a described action.

It is emphasized that the terms "comprises" and "comprising", when used in this
25 application, specify the presence of stated features, integers, steps, or components and do not preclude the presence or addition of one or more other features, integers, steps, components, or groups thereof.

The particular embodiments described above are merely illustrative and should
30 not be considered restrictive in any way. The scope of the invention is determined by the following claims, and all variations and equivalents that fall within the range of the claims are intended to be embraced therein.